

THE CEBAF CONTROL SYSTEM ARCHITECTURE

Rolf Bork
Continuous Electron Beam Accelerator Facility
12070 Jefferson Avenue
Newport News, VA 23606

ABSTRACT

The focus of this paper is on CEBAF's computer control system. This control system will utilize computers in a distributed, networked configuration. The architecture, networking and operating system of the computers, and preliminary performance data are presented. We will also discuss the design of the operator consoles and the interfacing between the computers and CEBAF's instrumentation and operating equipment.

INTRODUCTION

The Continuous Electron Beam Accelerator Facility (CEBAF) is a 4.0 GeV multipass accelerator, construction will commence in 1987. CEBAF will employ superconducting RF technology to facilitate electron beam acceleration. 400 RF klystrons, in twin linacs, will recirculate the beam in a one mile racetrack configuration. Before reaching the target end stations, the beam will pass through each linac four times. Over 1800 magnets will be used to transport the beam, providing steering, focusing, separation and recombination of the beam from the injector to the targets. Four beams will reside in each linac at one time. In addition to these systems, CEBAF will have a 2K refrigeration plant to provide liquid helium to the superconducting RF cavities in the linacs, and to the superconducting magnets in the end stations. The CEBAF control system will be tasked to operate and control these systems, along with various vacuum and safety systems.

SYSTEM ARCHITECTURE

The architecture of the control system is developed as a hierarchial, distributed computer network. First, the accelerator is split into six subsystems for control; those being safety, cryogenic/vacuum, injector, RF, beam transport, and beam switchyard. Each system is to have its own computers, assigned primarily to operate and control only equipment within that system.

Next, control functions are divided into two levels, known as supervisor and local. Supervisory computers are to provide overall system control, with facilities for operator interfaces. The local computers are to provide monitoring and control of system equipment within a geographic area of the machine. The local controls will also provide the necessary interface between the computers and the operational equipment. Figure 1 depicts a block diagram overview of the system.

SUPERVISOR CONTROL LEVEL

Each system has a console in the Accelerator Main Control Room (AMCR). The components which make up a console are depicted in Figure 2. Each console has its own computer, with graphics terminals and monitors, operator interfaces, and disc drive. These console computers represent the highest level of control in the hierarchy, and combined are known as the supervisor level.

Along with the six supervisor consoles, there will be a seventh, known as the I&C supervisor console. This console is identical to the others, but its computer will not be tasked with control of a particular subsystem. Its primary function is to provide displays of overall CEBAF operation, and provide a gateway to CEBAF's central computing facilities. It will also be available as a "hot" spare should one of the other supervisor computers fail. Modelling codes for overall CEBAF control will probably reside in this computer, although all supervisor computers would have this capability.

As the architectural design of the supervisory level developed, the architecture of computers at this level also was decided. These computers will employ three separate central processing units (CPU's), tightly coupled on a common memory bus with shared memory capability. One CPU is assigned to operator interface tasks, one to I/O and database management tasks, and the third to all other program tasks. This architecture was chosen to ensure consistent computer response to the operator and updating of the database from other

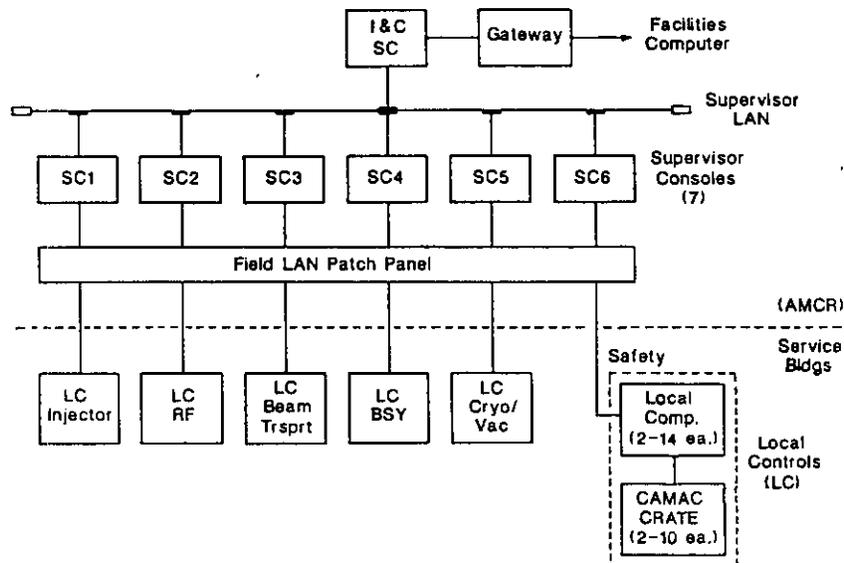


Figure 1 - Control System Architecture

computers within the control system, regardless of the number of programs currently running on the machine. Therefore, modelling codes, housekeeping, and various specific codes are shared by the third CPU, in no way impeding the performance of the other two tasks.

LOCAL LEVEL CONTROLS

Local control computers and their interfaces to the the operating equipment will reside in various service buildings placed strategically around the tunnel. Each of the six control subsystems previously mentioned will have from two to fourteen local computers, situated to operate and control equipment assigned to its subsystem in a given geographical location. The systems will consist of a computer and disc drive, from which it will automatically load and begin running its operating system and applications programming. No terminals will be permanently attached to the computers, but will have provisions for terminal connection for local testing and troubleshooting purposes.

Between these computers and the actual operating hardware, the interface will be via Computer Automated Measurement and Control (CAMAC) crates and associated CAMAC modules. The tie from computer to CAMAC crates will be via General Purpose Interface Bus (GPIB). Depending on the particular computer control location, a local computer will be connected to at least two and up to ten crates.

INTERCOMPUTER COMMUNICATIONS

As depicted in Figure 1, the computers of the control system are connected via an ethernet (IEEE-802.3) Local Area Network (LAN). There are seven LAN's in total. Six LAN's connect the local computers back to the AMCR, one dedicated to each control subsystem. The resulting parallel paths were chosen in order to maximize transmission speed from the local control level back to the supervisory level. This spreads out the data traffic and keeps the LAN from being a control system bottleneck. A standard operating system was also desired which supported LAN and was available from several vendors.

From the LAN patch at the AMCR, each line is connected to one of six supervisor console computers. In this manner, any console in the AMCR can be assigned to any system, and, in the event of a supervisor computer failure, another computer can be assigned to that subsystem by switching the LAN at the patch panel and loading in the necessary software.

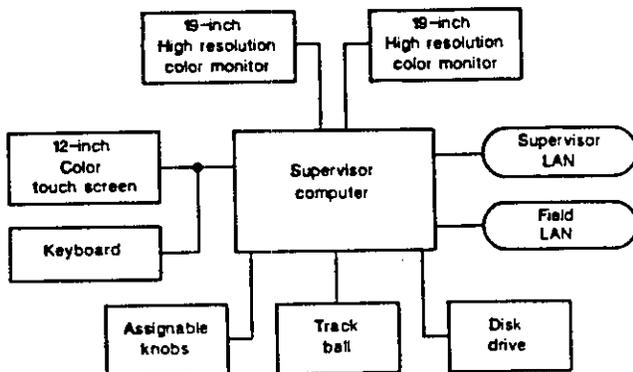


Figure 2 - Supervisor Console Block Diagram

Once the LAN is connected and software loaded, a supervisor console is tasked primarily to oversee the operation of its assigned system. To perform this function, it will be necessary for the supervisor computers to obtain certain information from other systems. It may be desirable to operate a system from other than its assigned supervisor console. An overall control system database must be maintained in each for use by modelling codes such that a modelling code running in one supervisor computer has control capability of all control subsystems. To facilitate this, all supervisor computers will be interconnected via another LAN, known as the supervisory LAN. Each supervisor computer will continuously broadcast updates on its system data to the other supervisor computers across this network.

PRELIMINARY SYSTEM TESTS

While the first procurements for the control system are just now being placed, several tests were run over the past year simulating various sections of the control system. Since any control system is primarily I/O intensive, testing is centered around this aspect. In the CEBAF control scheme, this involved mainly intercomputer communication and I/O between the local computers and CAMAC.

Ethernet was chosen for data transmission between computers in keeping with the use of industry standards as much as possible. Many commercially available UNIX operating systems provide drivers into this LAN, and with the growing number of vendors supporting UNIX and its standardization, UNIX and UNIX based machines were tested for suitability in the CEBAF control system. While UNIX is not known as a real time operating system, more real time hooks are being supplied by various vendors, and several real time UNIX systems are offered. The CEBAF design of the control system places much of the real time tasks in hardware, particularly in certain systems which require a decision and response in <10 μ sec. Also, the control system will not heavily rely upon interrupt driven I/O.

The test of the LAN was important in determining actual data throughput between computers to be configured at CEBAF. While IEEE802.3 ethernet is run at 10 Mbits/sec., the control system structure and data formatting places certain restrictions on the system. The control system must transfer packets of data from memory in one computer to memory in another. Data transmission size will normally be small, one message per packet of data (1400 Bytes) at a given time. Data transmission will be between supervisor computers and locals only (i.e., local computers will not communicate with each other directly). The object of this test was to determine actual data throughput, with all of the programmatic overhead included.

For this test, programmatic control of the LAN was through the use of link level 2 access (ISO model). This was done to reduce the overhead time involved with the upper layers of the LAN protocols. The drivers were supplied with the UNIX operating system. Due to limited equipment, the setup used one supervisor computer connected to two local control computers. The two local control computers were similar in structure to those to be employed in the control system, but their CPU ratings were 2.5 times slower.

For the first part of the test, data traffic was only between the supervisor computer and one local control computer. The supervisor computer loaded and transmitted single packets (1400 Bytes) of data to the local control computer, which would load the data into memory, pull the response data, and transmit a single packet back to the supervisor computer. This is similar to what will occur in the actual control system; the supervisor computer sending setpoints and other data in a single packet to the local computer, and the local computer responding with a packet of data, encoded with system status and operating points. In this fashion, local control computers will transmit data only when requested to do so by the supervisor computer.

In the first case, both computers transmitted and received an average 42 packets/second, or approximately 60K Bytes/sec. A monitoring program reported CPU usage in the local machine ran about 8%, with the supervisor CPU, a faster unit, reporting approximately 4% usage.

In the second part of the test, data was run to both local control computers. Using the same operation as in part one, the data transfer rate at the locals remained the same, approximately 42 packets/sec send and receive, with data at the supervisor computer now double, approximately 84 packets/second, CPU usage on the supervisor computer was now up to approximately 8%.

The goal of this configuration was to be able to update the entire database within a system every 100 msec. This test showed the capability of achieving an update in approximately 25 msec. Therefore, if the data extrapolates with a full complement of local computers on the LAN, this scheme should provide the desired results.

The second major communications link is between the local computers and the CAMAC equipment. Once again, a standard was chosen, in this case GPIB. This interface is supported by many vendors and drivers are available in UNIX.

Tests were run on this communication link to determine:

- 1) What is the maximum data transfer rate available
- 2) What access time can be expected to CAMAC crates in the control system

In the control system, most CAMAC crate data transfers involve data amounts <320 Bytes. There may also be instances where larger data transfers are involved, as with crates in which waveform digitizers are installed. Both data burst speeds and access time overhead were of interest.

For the test setup, a computer similar to the type to be incorporated as a local computer (although CPU performance on this unit is a factor of 2.5 times slower) was connected via GPIB to a high speed GPIB CAMAC crate controller. Various modules were placed within the crate, each with read and write data capabilities. All software was written in C and run under UNIX, using the supplied UNIX drivers.

To test burst data transfer rates, blocks of 30,000 data bytes were written to and read from the crate. Timing runs showed data transfers ran approximately 250 KBytes/sec. This

test was also run with a computer of the appropriate supervisor level type, with results of 580 KBytes/second.

Of perhaps more interest from the control system viewpoint is the crate access time when small amounts of data are to be transferred. The C program for this test would read and write 64 bytes of data to CAMAC modules under address scan mode. In this test, timing runs showed a complete cycle of writing 64 Bytes and reading back 64 Bytes took an elapsed time of 4 msec. Since I/O for this test was primarily under CPU control, the faster computers to be actually installed in the control system should reduce this run time.

Along with these I/O performance tests, various other tests were run using UNIX. These included interactive graphics capabilities, and real time control using ladder logic and PID control loop applications. The graphics drivers for generating displays and handling input from operator interfaces proved easy to implement and use. Real time control also showed satisfactory results, within several msec. resolution of closed loop controls, even when the computer was multitasked with interactive graphics and other programming.

SUMMARY

With the successful results of the tests previously described, the implementation of the control system is beginning to take place. Computer equipment and peripherals for two supervisor consoles and six local control computers should be available by April of 1987. Plans are to use these first systems to develop and refine the networking software and computer/CAMAC communications. This should be in place for testing, on the equipment available, by late summer of 1987.

ACKNOWLEDGMENT

This work was supported by the Department of Energy under Contract DE-AC05-84ER4015.